DISEASE PREDICTION USING MACHINE LEARNING

A PROJECT REPORT

Submitted by

BRINDA M	**(2011011)**
GOWTHAM A	**(2011015)**
SASMITA M	**(2011041)**

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

SRI RAMAKRISHNA ENGINEERING COLLEGE

[Educational Service: SNR Sons Charitable Trust]
[Autonomous Institution, Reaccredited by NAAC with ‘A+’ Grade]
[Approved by AICTE and Permanently Affiliated to Anna University, Chennai]
[ISO 9001:2015 Certified and all eligible programmes Accredited by NBA]
VATTAMALAIPALAYAM, N.G.G.O. COLONY POST, COIMBATORE – 641 022.

COIMBATORE-641 022

ANNA UNIVERSITY: CHENNAI 600 025

MAY 2022

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

16IT266 PROJECT WORK

Certified that this project report **“DISEASE PREDICTION USING MACHINE LEARNING”** is the bonafide work of “**BRINDA M (2011011), GOWTHAM A (2011015), SASMITA D (2011041)”** who carried out the project work under my supervision.

SIGNATURE	**SIGNATURE**
Dr. M. Senthamil Selvi	Mr. M. Logaprakash
HEAD OF THE DEPARTMENT	**SUPERVISOR**
Professor	Assistant Professor / AI&DS
Department of Information Technology	Department of Information Technology
Sri Ramakrishna Engineering College	Sri Ramakrishna Engineering College
Coimbatore-641022	Coimbatore-641022

Submitted for the Project Viva-Voce examination held on _____________

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

We thank the almighty for his blessings on us to complete this project work successfully.

With profound sense of gratitude we thank the Management, Managing Trustee**, Thiru. D.Lakshminarayanaswamy** and Joint Managing Trustee, **Thiru. R. Sundar** for having provides us the necessary infrastructure required for the completion of our project.

With profound sense of gratitude, we sincerely thank the **Head of the Institution**, **Dr. N.R. Alamelu for** her kind patronage, which helped in pursuing the project successfully.

With immense pleasure, we express our hearty thanks to **Head of the Department, Dr. M. Senthamil Selvi** for her encouragement towards the completion of this project.

We thank our Academic Coordinator **Dr.J. Anitha , Associate Professor,** Project Coordinator **Mrs. S. Jansi Rani, Associate Professor (SR.GR),** Department of Information Technology and our Project guide, **Mr.M. Logaprakash** for their encouragement towards the completion of this project.

We convey our thanks to all the teaching and non-teaching staff members of our department who rendered their co-operation by all means for completion of this project.

ABSTRACT

Humans are now threatened with a variety of diseases because of the current state of the environment and their lifestyle choices. It is important to recognize and predict such diseases at an early stage in order to prevent them from progressing late. Most of the time, doctors struggle to accurately identify diseases by hand. This is accomplished by the use of Machine learning algorithm. The intended outcomes and scope of the project is to predict the disease so that patients could, receive early treatment, lowering the risk of mortality and saving patients' lives, and the cost of disease treatment can be reduced to some extent by early identification. It is also used to predict diseases based on a patient's symptoms.

TABLE OF CONTENTS

APPENDICES

LIST OF FIGURES

LIST OF ABBREVATIONS

AI	Artificial Intelligence
ML	Machine Learning
CNN	Convolution Neural Networks
KNN	K-Nearest Neighbor
CD	Chronic Disease
XGB	Extreme Gradient Boost
CHT	Circular Hough Transform

CHAPTER 1
INTRODUCTION

When a person currently suffers from a certain disease, he or she must seek medical advice, which is both times demanding and costly. Furthermore, if the user is out of reach of doctors and hospitals, the sickness may be difficult to identify. So, if the above-mentioned procedure can be accomplished using an automated software that saves time and money, it may be simpler for the patient, making the process easier.

Disease Predictor is a stand-alone application that anticipates the user's disease supported the symptoms inputted. The Disease Prediction System comprises data sets gathered from several health-related websites. The user will be able to determine the likelihood of the disease based on the symptoms provided by Disease Prediction system

Doctors sometimes struggle to appropriately identify diseases by hand. A machine learning method might be used to do this. Because it is used to provide a prognosis for the patient, it can help doctors provide diagnoses more swiftly. People can benefit from this approach since they always have their device with them to know about the disease

1.1 Machine Learning

Machine learning comes under the domain of artificial intelligence (AI). In general its aim is to understand the structure of data which can be put it into models that man can understand and use. Machine learning is considered as a type of computer science, it is marked from traditional computer science methods. An algorithm is a set of explicitly programmed instructions that a computer uses to calculate and find the solution for the problem. This algorithm, on the other hand , allows a computer to train a data input and then it uses statistical analysis that helps to generate values within a particular division . So, this learning technique allows computers to create models from the set of sample data that aids to automatically make decisions based on the input data. This learning has also given benefits to all technology user today. Social media platforms can use facial recognition technology help users to tag and send photos of friends. Optical character recognition (OCR) technology has been used to transform the text images to the movable or animated type. Machine learning-powered recommendation engines gives the recommendations of what movies or television shows to see based on the user preferences. Self - driving cars may soon be able to come to the role and might influence the customers. Machine learning is now evolving rapidly. Finally, there are things to keep in mind when it comes to working with machine learning methodologies or analysing the impact of this learning processes

1.2 Machine Learning Methods

Assignments are for the most part grouped into general classes in AI. These classes depend on how learning is gotten or input on learning is given to the created framework.

The two most widely used machine learning methods are unsupervised learning, which works with unlabelled data to find structure within that input data, and supervised learning, which trains algorithms for example input and output data that has been labelled by humans.

1.2.1 Supervised Learning

The computer is provided with some examples of inputs labelled with the desired outputs in machine learning. The goal is to make the algorithm to learn by comparing the actual output with the taught output in order to detect errors and to modify the model. Finally, supervised learning makes use of patterns to forecast label values on unlabelled data.

For example, in supervised learning an algorithm may be show data containing sharks labelled as fish images, oceans labelled as water images. The regulated learning calculation ought to have the option to distinguish unlabelled shark pictures as fish and unlabelled sea pictures as water in the wake of being prepared on this information. A frequent use of supervised learning is the use of historical data to anticipate statistically likely future events. Photos if dogs which are untagged can be used as input data in supervised learning to classify tagged photos of dogs

1.2.2 Unsupervised Learning

The learning algorithm is left to find commonalities among its input data because the data in unsupervised learning is unlabelled. Machine learning approaches that promote unsupervised learning are especially beneficial since unlabelled data is more abundant than labelled data. The simple goal in unsupervised learning may have to find hidden patterns within a dataset, but the feature learning may also be its goal which provides the computational machine to discover automatically the representation needed to classify raw data.

This learning is usually used for transactional data. You may have a big data set of customers and their purchases, but as a human, you're unlikely to be able to figure out what related features can be derived from customer profiles and purchase kinds. With this data given into an unsupervised learning algorithm, it may be possible to determine that women of certain age group who purchase are unscented soaps are likely to be pregnant, and thus a marketing campaign promoting pregnancy and baby products are targeted at this group to increase their purchases

1.3 Approaches

Statistics is beneficial for understanding and leveraging machine learning algorithms since it is related to computational statistics. So is becomes essential to have a background in statistics.

1.3.1 Decision tree

Decision trees are supervised learning techniques which can be used for both classification and regression problems but it is commonly used for classification . It is a tree classifier , where the nodes represent the characteristics of data set , decision rules are represented by the branches , and outcome represented by each lead node .

In decision tree there are two nodes : Decision node and the tree node . Decision nodes makes any decision and also have multiple branches , whereas the leaf nodes are the output of the decisions and it does not have further branches . The features of the given dataset are often used to make judgments or run tests. All viable solutions to a problem/decision are graphically displayed under particular circumstances.

It's called a decision tree because it's structured like a tree, with the root node at the top and branches outwards to form a tree-like structure.

1.3.2 Random forest

Random forest is well-known algorithm in machine learning from the supervised learning technique. It can be applied to both classification and regression problems in machine learning. It really is based on the concept of ensemble learning, which is a process in which multiple classifiers are combined to solve a complex problem and improve the model's performance. As the name implies, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Random Forest, rather than relying on a single decision tree, takes predictions from each tree and predicts the final outcome based on a majority of prediction votes. The greater number of trees in the forest, the higher the accuracy and the lower the risk of overfitting

1.3.3 Naïve Bayes

This algorithm is a supervised learning algorithm which uses Bayes theorem to solve classification problems. It is used with large training dataset in text classification.

The Naive Bayes classifier is simple and effective classification algorithm that focus on the development of fast machine learning methods that is to make quick predictions. It is a probabilistic classifier, which means it predicts based on an object's probability. Spam filtration, sentiment analysis, and article classification are some of the popular applications of naive Bayes algorithm

1.3.4 K-Nearest Neighbour

Under Supervised learning techniques, K-Nearest Neighbour comes as a straightforward and uncomplicated Machine learning algorithm. Resemblance betwixt existing case and new data are checked. It places the latest case in the group which most resembling to the old groups.

The K-Nearest Neighbour reserves every obtainable data with it. Then it checks the resemblance between them to categorise latest data points, which enables K-Nearest Neighbour algorithm to fastly group the latest data points into convenient categories.

For both Classification and Regression, K-Nearest Neighbour algorithm can be utilized. For classification only it is used most often. There is no presumption is made about the used data, which makes the K-Nearest Neighbour algorithm a non-parametric algorithm.

1.4 Overview of the Project

This project described a symptom-based disease identification method. Therefore, it can help to make good decisions and predict the right output using a large amount of training data. To detect the disease, many symptoms are used as input. The frequency of symptoms in line with disease is utilized to predict the disease. To obtain the highest accuracy, massive amounts of training data are given. This project aims to forecast diseases more accurately using machine learning approaches. The symptoms are used as attributes in this system to forecast diseases using the Decision tree classifier method, RandomForest algorithm, and Naïve bayes algorithm.

CHAPTER 2

LITERATURE SURVEY

Various studies on disease prediction systems employing deep learning and machine learning have been conducted. This research is conducted prior to beginning the project and understanding the various strategies that have previously been employed. This research assisted in determining the upsides and downsides of the existing system.

2.1 Dhiraj Dahiwade; Gajanan Patle; Ektaa Meshram in "Designing Disease Prediction Model Using Machine Learning Approach"

This research proposed a prediction system for general diseases based on machine learning algorithm. It used algorithms like KNN and CNN to categorize individual data since the medical records are expanding rapidly that can be processed in order to forecast precise diseases on the basis of the symptoms. It has accurate results in disease prediction as an output where patients' data is taken as an input, which let us to grasp the degree of how the disease is predicted. By using this approach, disease prediction and risk prediction may take less time and expense. It compares the outcomes of the KNN and CNN algorithms in terms of accuracy and time, and the accuracy of the CNN method is found to be higher than that of the KNN algorithm, and the time required to classify for CNN is comparatively less.

2.2 G. Battineni, G. G. Sagaro, N. Chinatalapudi, and F Amentain in "Applications of machine learning predictive models in the chronic disease diagnosis"

The current study assessed studies related to the diagnosis of chronic disorders. However, accurate technique implementation or model selection is required to make optimal conclusions, since recent studies suggest that some ML models are growing malicious data with the already available confined datasets, which might have serious consequences. Diagnostic restrictions can also lead to threatening attacks, and, in rare cases, it constitutes a fatal factor. In contrast, incorrect diagnosis causes distrust in machine learning use, which might drive policymakers to avoid using prediction models. As a result, assessments of prediction models can give evidence for effective approaches for CDs diagnosis. AI techniques such as machine learning, and deep learning may play an important role to interpret certain chronic illnesses in the future. This also includes cognitive computing. However, predictive model approaches are increasingly attracting scholars in health care field. New developments in medical care are made and that gives access to digital data is expanded, and also avenues for productivity and decision support gains open up. Thus, the models are intended to captivate the importance to look after the patients and also aids in lowering medical costs.

2.3 Shanjida Khan Maliha, Romana Rahman Ema, Simanta Kumar Ghosh,Helal Ahmed, Md. Rafsun Jony Mollick in “Cancer Disease Prediction Using Naive Bayes, K-Nearest Neighbor and J48 algorithm”

This report is based on a medical research dataset that can predict cancer sickness. According to the findings, the majority of cancers are caused by cigarette use. contingent upon the dataset it used almost 3 classification techniques. In this research, it used weka tools for classifying. Weka includes features such as preprocessing, classification, grouping, regression, and visualisation. Because it contains no personal information, the dataset does not incite violence. It only contains medical information. To determine the confusion matrix, three approaches are utilised. The classification algorithm's outcome is given by the confusion matrix. It includes information on both the actual and expected categorization. This report had almost taken 10 different classes.

Cancer, esophageal cancer , Prostate Cancer, Blood Cancer, Breast Cancer, no cancer , Brain cancer, Ovarian cancer, Lung cancer, Pancreatic Cancer, colorectal, This is used to finalize the accuracy, specificity, F-score, error rate, precision and sensitivity. Table is designed to be easily understood. It compares three categorization algorithms: K-Nearest Neighbour, algorithm j48, and Naive bayes algorithm. The correlation table makes it apparent which of these assortment model is exceptional in performance. The functionalities of all 3 algorithm was worked well, whilst of comparing them all, K-Nearest Neighbour outperforms both in the terms of accuracy

2.4 K Prajwal; Tharun K; Navaneeth P; Anand Kumar M in "Cardiovascular Disease Prediction Using Machine Learning"

This paper demonstrates many automated computational Cardiovascular Disease Prediction procedures that may be accomplished using Supervised Learning, Classification, and Regression methods. Various characteristics are used to test the algorithms. The suggested method's purpose is to provide accurate disease prognosis. The decision classifier technique was found to be quite effective in predicting disease including characteristics such as age, BMI, cholesterol, and others. The addition of the BMI feature enhanced prediction accuracy. As a consequence of analysing the results, the proposed method produces a more precise forecast of cardiovascular illnesses. The Decision Tree method was discovered to be more efficient and tested with the best accuracy. The XGB classifier was used to determine the significance of each characteristic in the prognosis of heart disease.

2.5 Muhammad Bilal Butt, Majed Alfayad, Nouh Sabri Elmitwally in "Diagnosing the Stage of Hepatitis C Using Machine Learning"

In advance to the improvement in the medical field by the machine learning, the patient had to go through many different variety of tests by the doctor to identify the stage of the Hepatitis C. Many useless, unnecessary tests are undergone by the patient while the diagnostics process is under review to finalise the disease stage. It is consumes both time and money for the patient. To avoid unnecessary check-ups and to reduces the wasted time for both patient and the doctor a interrogatory test should be conducted in order to find the stage of Hepatitis c disease that is present in the patient which will be valuable information fo the patient to begin the treatment procedures. IHSDS which was proposed is validated after training the datatset, from with the training phase gives a accuracy of 98.89 percentange and the validation phase gives a accuracy of 94.44 percentage. The failure rate and the accuracy rate is produced by comparing the previous model with the proposed model, demonstrating that the new model outperforms previously known approaches.

CHAPTER 3

PROJECT DESCRIPTION

3.1 PROBLEM DEFINITION

The health industry has reached billions of dollars. Every day, the healthcare industries has been generating vast amounts of healthcare data, which can be used to extract insights where it can predict a patient's future illness based on treatment history. treatment and healthcare data. In the future, all these data hidden in this health data will be used to make emotional decisions about a patient's health. In addition, this field can be improved by using informational data in the medical field.

3.2 OBJECTIVE

- Disease prediction using machine learning is a system that predicts diseases based on symptoms provided by patients or other users. The algorithm takes the symptoms reported by a single user as input and outputs a likelihood of their disease.
- It is critical to diagnose and forecast such diseases in their early stages to avoid them developing to their late phases. Most physicians fail to precisely diagnose diseases by hand, thus our technology helps reduce such difficulties.
- It is not feasible to correctly monitor patients every day in all circumstances, and consultation with a doctor for 24 hours is not available since it demands more intelligence, time, and competence. As a result, we employ this disease prediction technology.
- Doctors' consultation fees are expensive; our approach can reduce the overall cost of consultation.

3.3 BLOCK DIAGRAM

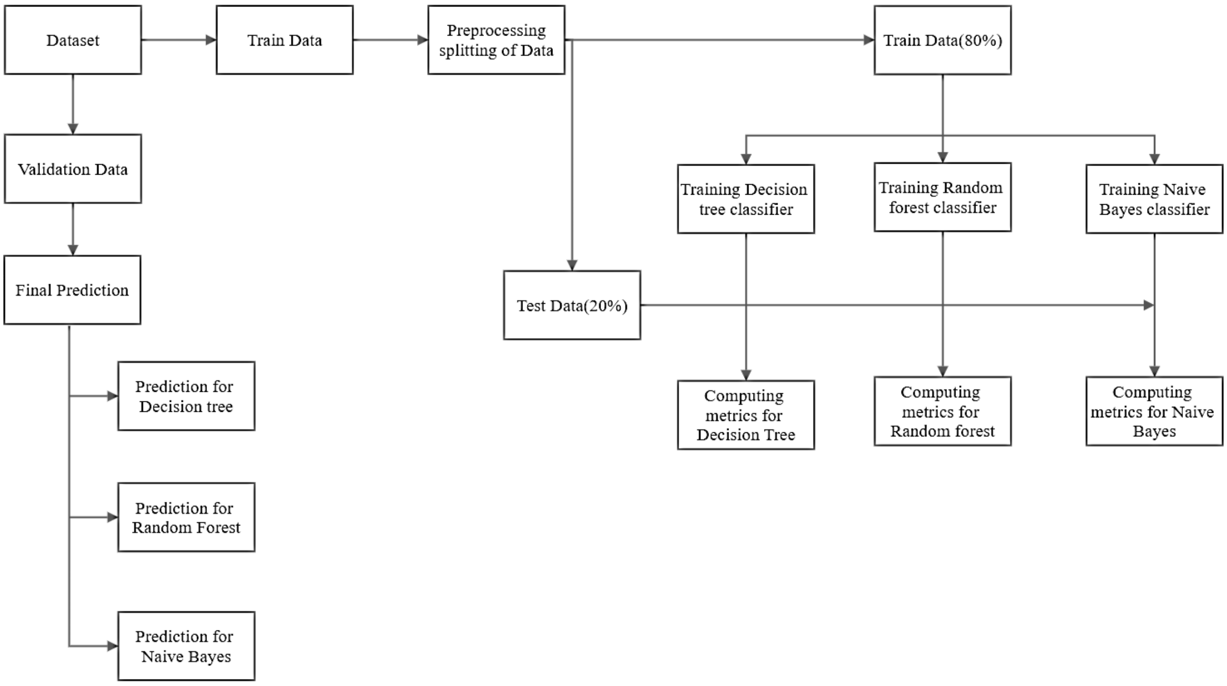

Dataset
Train Data
Preprocessing splitting of Data
Train Data(80%)
Validation Data
Training Decision tree classifier
Training Random forest classifier
Training Naive Bayes classifier
Final Prediction
Test Data(20%)
Prediction for Decision tree
Computing metrics for Decision Tree
Computing metrics for Random forest
Computing metrics for Naive Bayes
Prediction for Random Forest
Prediction for Naive Bayes

Fig 3.3.1 Block Diagram

3.4 MODULES

3.4.1 Decision Tree

For both Regression and classification, a supervised learning technique in machine learning is used, which is Decision tree. The feature of the dataset is denoted by the internal nodes, the outcome is denoted by the every leaf nodes, and the decision rules are indicated by the branches.

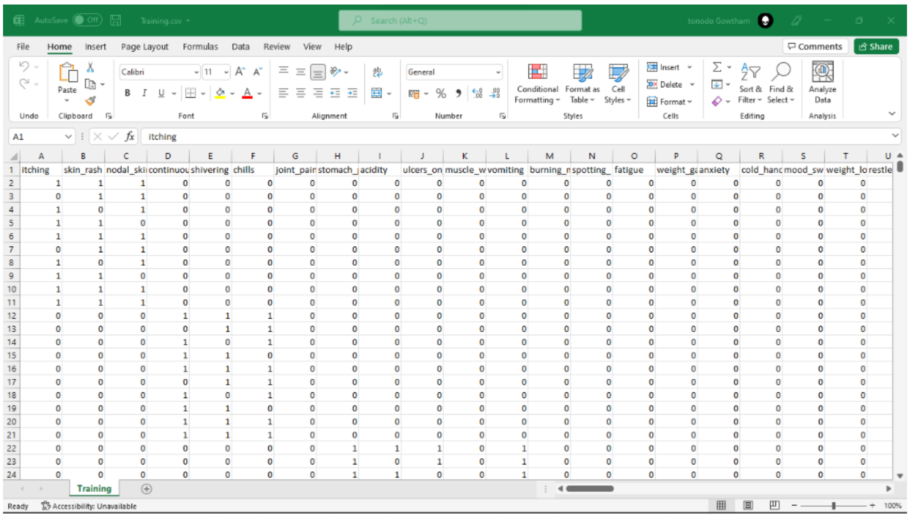
This proposed system would predict diseases by implementing Decision Tree Classifier. This approach uses the processes below

1) Gathering the Data
2) Cleaning the Data
3) Model Building

4) Inference

❖ Gathering the Data

In machine learning, data preparation is the most critical and 1st step in solving the problem. For this project, we will use a Kaggle website to download dataset. Two CSV files are divided from the dataset , where training and testing files respectively. The dataset has 130 columns, with 129 columns representing symptoms and the last column representing the prognosis.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	itching	skin_rash	nodal_ski	continuou	shivering	chills	joint_pain	stomach_	acidity	ulcers_on	muscle_w	vomiting	burning_m	spotting_	fatigue	weight_ga	anxiety	cold_hanc	mood_sw	weight_lo	restle
2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
3	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
4	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
5	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
6	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
7	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
8	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
9	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
11	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
12	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
13	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
14	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
15	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
16	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
17	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
18	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
19	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
20	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
21	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
22	0	0	0	0	0	0	0	1	1	1	0	1	0	0	0	0	0	0	0	0	
23	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	
24	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0	0	0	0	

Fig 3.4.1.1 Dataset

❖ Cleaning the Data

In the machine learning projects the most critical part is cleaning. The machine learning model's quality is determined on the quality of our data. All of the columns in our dataset are numerical, except for the last column, prognosis, which is a string that has been encoded to numerical form.

❖ Model Building

We split the data after gathering and cleaning it so that we could work on the modelling phase. The data is ready for use in training a ML model. Model will use this cleaned data for training the decision tree Classifier for which it will be used to build the model.

- **Inference**

Post training of the decision tree classifier models, it will forecast the illness based on the input symptoms. The prediction are made more robust and accurate.

3.4.2 RandomForest

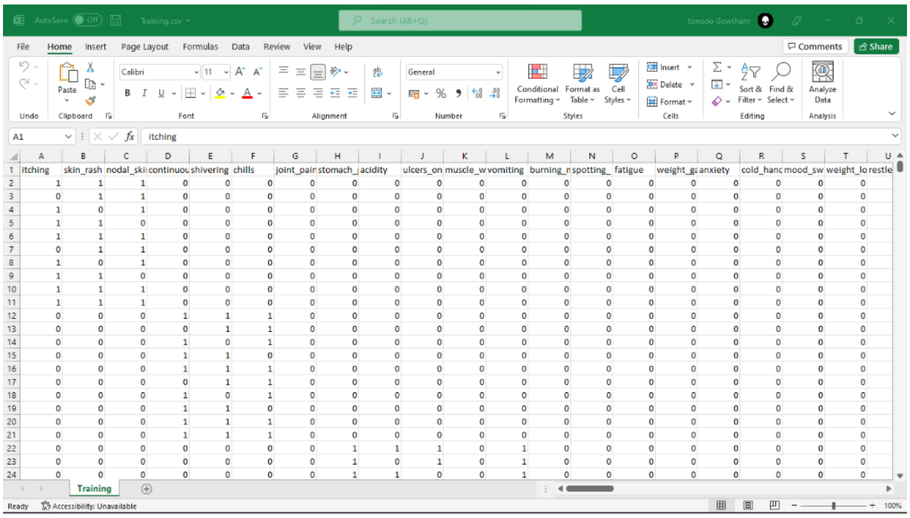
Random Forest is also one of the popular algorithm in machine learning that comes under the technique called supervised learning. It is used for both Regression problems and classification in ML. It adheres on ensemble learning, where multiple classifiers are combined to solve huge and complicated problems that aids in the improvement in the performance of a model. This algorithm contains a number of decision trees as subsets of the given dataset by which it takes the average of the subsets to improve or predict the accuracy for the given that dataset.

This proposed system would predict diseases by implementing Decision Tree Classifier. This approach uses the processes below

1) Gathering the Data
2) Cleaning the Data
3) Model Building
4) Inference

- **Gathering the Data**

The first step to solve any machine learning problem is data preparation. For this project, we will use a Kaggle website to download dataset. This dataset is divided into two CSV files, one for training the dataset and the other for testing. The dataset has 130 columns, with 129 columns representing symptoms and the last column representing the prognosis.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	itching	skin_rash	nodal_ski	continuou	shivering	chills	joint_pain	stomach_	acidity	ulcers_on	muscle_w	vomiting	burning_m	spotting_	fatigue	weight_ga	anxiety	cold_hanc	mood_sw	weight_lo	restle
2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
3	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
4	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
5	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
6	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
7	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
8	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
9	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
11	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
12	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
13	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
14	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
15	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
16	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
17	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
18	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
19	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
20	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
21	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
22	0	0	0	0	0	0	0	1	1	1	0	1	0	0	0	0	0	0	0	0	
23	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	
24	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0	0	0	0	

Fig 3.4.2.1 Dataset

❖ Cleaning the Data

The important critical phase in any project involving the use of machine learning is data cleaning. The machine learning model's quality is determined on the quality of our data. All of the columns in our dataset are numerical, except for the last column, prognosis, which is a string that has been encoded to numerical form.

❖ Model Building

We split the data after gathering and cleaning it so that we could work on the modelling phase. The data is ready for use to train a machine learning model. Thus this cleaned data is used to train the decision tree Classifier that will be used to build the model.

❖ Inference

Post training of the decision tree classifier models, it will forecast the illness based on the input symptoms. The prediction are made more robust and accurate.

3.4.3 Naïve Bayes

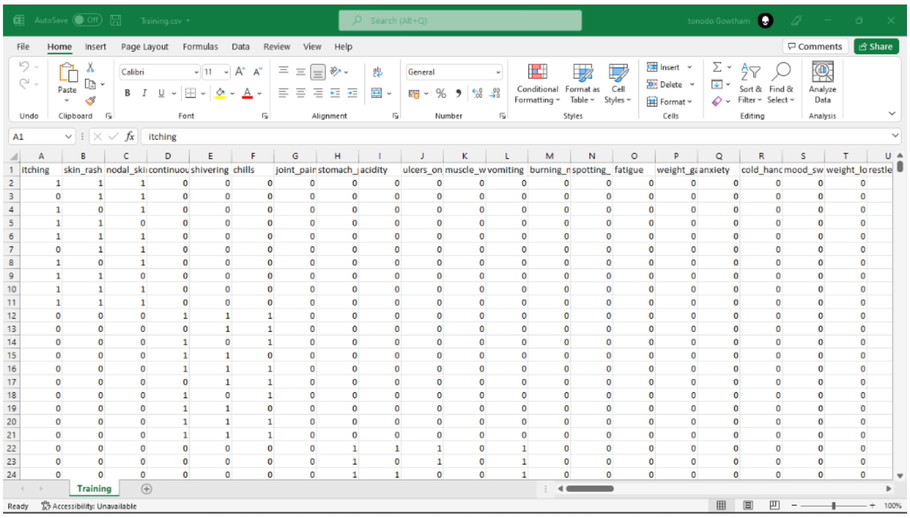
This algorithm is a supervised learning algorithm, which is typically based on Bayes theorem and is used for solving higher classification problems.

This proposed system would predict diseases by implementing Decision Tree Classifier. This approach uses the processes below

1) Gathering the Data
2) Cleaning the Data
3) Model Building
4) Inference

- **Gathering the Data**

The first step to solve any machine learning problem is data preparation. For this project, we will use a Kaggle website to download dataset. This dataset is divided into two CSV files, one for training the dataset and the other for testing. The dataset has 130 columns, with 129 columns representing symptoms and the last column representing the prognosis.



itching

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	itching	skin_rash	nodal_ski	continuou	shivering	chills	joint_pain	stomach_	acidity	ulcers_on	muscle_w	vomiting	burning_m	spotting_	fatigue	weight_ga	anxiety	cold_hanc	mood_sw	weight_lo	restle
2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
3	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
4	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
5	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
6	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
7	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
8	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
9	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
11	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
12	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
13	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
14	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
15	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
16	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
17	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
18	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
19	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
20	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
21	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
22	0	0	0	0	0	0	0	1	1	1	0	1	0	0	0	0	0	0	0	0	
23	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	
24	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0	0	0	0	

Fig 3.4.3.1 Dataset

❖ Cleaning the Data

The most critical phase in a machine learning project is cleaning. The machine learning model's quality is determined on the quality of our data. All of the columns in our dataset are numerical, except for the last column, prognosis, which is a string that has been encoded to numerical form.

❖ Model Building

We split the data after gathering and cleaning it so that we could work on the modelling phase. The data is ready for use to train a machine learning model. Thus this cleaned data is used to train the decision tree Classifier that will be used to build the model.

❖ Inference

Post training of the decision tree classifier models, it will forecast the illness based on the input symptoms. The prediction are made more robust and accurate.

3.4.4 Front-end

Tkinter is a standard Python library for designing graphical user interfaces (GUIs) for desktop applications. Building desktop applications using Tkinter is not a difficult endeavor.

Tk, Python's default GUI package, will be the major GUI toolkit we adopted. Tk will be accessed via its Python interface, Tkinter (short for Tk interface).

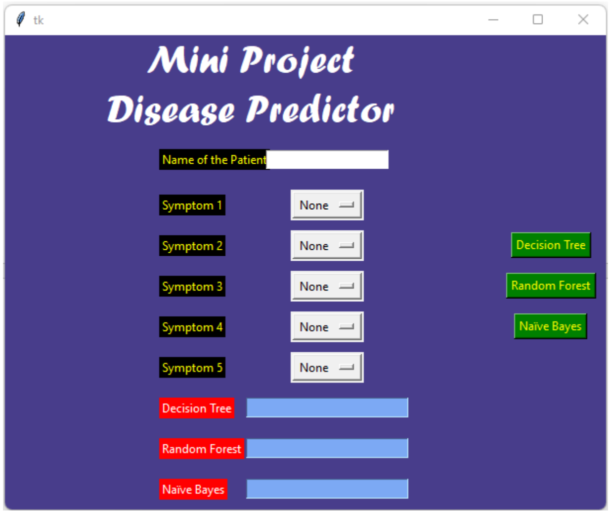


Fig 3.4.4.1 TKInter Output

3.5 Benefits:

- We can detect diseases in their early stages, allowing us to avert serious repercussions.
- It can be used to identify disease in those who do not have immediate access to a physician.
- It identifies typical sickness signs such as dizziness and stomach discomfort, and many others.
- More Personalized Medication.
- Less number of workers required.
- Because it is free to use, many users may benefit from it.

CHAPTER 4
RESULTS & DISCUSSON

This method's major goal is to predict general diseases. It can be oden by obtaining symptoms from the patient. A massive quantity of data is employed to train the model to prevent occlusion. This model outperforms other machine learning models in terms of accuracy. The Random Forest classifier, Decision Tree, Naive Bayes classifier were used to train the model. This dataset has 4920 distinct sets of disease symptom frequency. It is expected that the model will be able to classify with up to 95% accuracy.

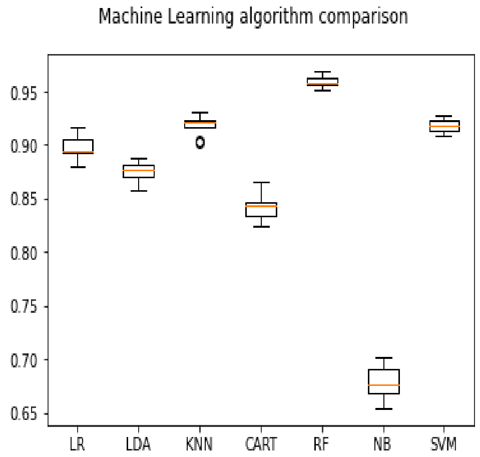


Fig 4.1 Machine Learning Algorithm

CHAPTER 5
CONCLUSION AND FUTURE SCOPE

CONCLUSION:

This project aims to predict disease from symptoms. This project is designed so that the system receives symptoms from the user as input and produces output. That is, it predicts disease This system provides a user-friendly environment and is easy to use. The system is based on standalone applications, so users can use it anytime. In summary, when modeling disease risk, the accuracy of disease prediction depends on the diversity characteristics of data. This systematic review aims to determine the performance of the limitations and the future use of the software in healthcare industry. Insight helps inform future developers that predictability of disease software and promote personalized patient care. Program predicts the disease of patient. Disease predictions are performed by the user icon. This system uses decision trees, and the Naive Bayes algorithm to predict diseases. For data format, the system uses machine learning algorithm database to data process. Machine Learning The skill was developed to successfully predict outbreaks. The predictions here use the Nave Bayes algorithm (model accuracy: 84.47%), decision tree (model accuracy: 91%), and RandomForest algorithm (model accuracy: 93%)

FUTURE SCOPE:

While disease prediction is now limited to a few diseases, many more disease prediction data sets can be added in the future. As a result, it improves prediction accuracy as well as disease prediction range. It can be expanded up quite large in the long term.

APPENDICES

APPENDIX 1

SOFTWARE DESCRIPTION

- **Python 3.9**

Python which has been considered as ahigh-level, interpreted, object-oriented scripting language is designed to be very readable and also it is dynamic language. It often uses English keywords while other languages use punctuation and it has less syntactic structure than other languages.To install the software

https://www.python.org/downloads/release/python-3910/

- **Jupyter Notebook**

Jupyter is one of the latest web-based interactive development environment for notebooks, code, and data. Jupiter has flexible interface that allows users to configure and arrange workflows in data science, scientific computing, computational journalism, and machine learning. A modular design invites extensions to expand and enrich functionality.

- **Python Libraries**

  - **Scikit learn**

  For Machine learning one among the most powerful, robust and useful library is Scikit-learn (Sklearn). Statistical modelling contains regression, classification, dimensionality reduction through a continuous interface in python, and clustering and also reliable tools for ML is provided in Scikit-learn (Sklearn) library. Scikit is written mostly in python, and building block of this library is matplotlib, SciPy and NumPy.

- **Pandas**

Pandas, a Python package that creates operating with relative or labelled information straightforward and intuitive. The goal of this library is to supply a basic framework for playacting real-world data analysis in Python. we try to be the foremost powerful and versatile open supply data manipulation and analysis tool out there altogether languages. The goal has been reached in a very manner that was hoped for.

- **Numpy**

A popular Python library that is used for working with arrays with functions that can be worked efficiently in the domain of linear algebra, fourier transform, and matrices is by the use of numpy, which was found in the year 2005 by Travis Oliphant. It is also considered as an open source where users can use if freely. The abbreviated form of numpy is Numerical Python.

- **Tkinter**

  The Tk GUI toolkit has a python binding called tkinter and which is the standard Python interface for this GUI toolkit. Standard GNU/Linux, Microsoft Windows and macOS installs of Python uses tkinter in their output library. Tk interface is otherwise known or so-called as tkinter.

HARDWARE REQUIREMENTS

- **Operating System: Windows 7 or Above**
- **RAM: Minimum of 4GB**

APPENDIX 2
SOURCE CODE

1) Disease Prediction using Machine Learning

```
# Importing Required Libraries for the project
# Importing numpy for managing operations in array
# Importing Pandas for data manupulation
# Importing tkinter for Frontend GUI

import pandas as panda
from tkinter import *
import numpy as nump

list11=['back_pain','constipation','abdominal_pain','diarrhoea','mild_fever','yellow_urine',
'yellowing_of_eyes','acute_liver_failure','fluid_overload','swelling_of_stomach',
'swelled_lymph_nodes','malaise','blurred_and_distorted_vision','phlegm','throat_irritation',
'redness_of_eyes','sinus_pressure','runny_nose','congestion','chest_pain','weakness_in_limbs',
'fast_heart_rate','pain_during_bowel_movements','pain_in_anal_region','bloody_stool',
'irritation_in_anus','neck_pain','dizziness','cramps','bruising','obesity','swollen_legs',
'swollen_blood_vessels','puffy_face_and_eyes','enlarged_thyroid','brittle_nails',
'swollen_extremeties','excessive_hunger','extra_marital_contacts','drying_and_tingling_lips',
'slurred_speech','knee_pain','hip_joint_pain','muscle_weakness','stiff_neck','swelling_joints',
'movement_stiffness','spinning_movements','loss_of_balance','unsteadiness',
'weakness_of_one_body_side','loss_of_smell','bladder_discomfort','foul_smell_
of urine',
'continuous_feel_of_urine','passage_of_gases','internal_itching','toxic_look_(typ
hos)',
'depression','irritability','muscle_pain','altered_sensorium','red_spots_over_body'
,'belly_pain',
'abnormal_menstruation','dischromic
_patches','watering_from_eyes','increased_appetite','polyuria','family_history','m
ucoid_sputum',
'rusty_sputum','lack_of_concentration','visual_disturbances','receiving_blood_tr
ansfusion',
'receiving_unsterile_injections','coma','stomach_bleeding','distention_of_abdom
en',
'history_of_alcohol_consumption','fluid_overload','blood_in_sputum','prominent
_veins_on_calf',
'palpitations','painful_walking','pus_filled_pimples','blackheads','scurring','skin_
peeling',
'silver_like_dusting','small_dents_in_nails','inflammatory_nails','blister','red_sor
e_around_nose',
'yellow_crust_ooze']

diseases=['Fungal infection','Allergy','GERD','Chronic cholestasis','Drug
Reaction',
'Peptic ulcer diseae','AIDS','Diabetes','Gastroenteritis','Bronchial
Asthma','Hypertension',
' Migraine','Cervical spondylosis',
'Paralysis (brain hemorrhage)','Jaundice','Malaria','Chicken
pox','Dengue','Typhoid','hepatitis A',
'Hepatitis B','Hepatitis C','Hepatitis D','Hepatitis E','Alcoholic
hepatitis','Tuberculosis',
'Common Cold','Pneumonia','Dimorphic hemmorhoids(piles)',
'Heartattack','Varicoseveins','Hypothyroidism','Hyperthyroidism','Hypoglycemia
','Osteoarthristis',
'Arthritis','(vertigo) Paroymsal  Positional Vertigo','Acne','Urinary tract
infection','Psoriasis',
'Impetigo']

list22=[]
for x in range(0,len(list11)):
    list22.append(0)

# disease prediction - test dataset dataframe
dataframe=panda.read_csv("Training.csv")



dataframe.replace({'prognosis':{'Fungal
infection':0,'Allergy':1,'GERD':2,'Chronic cholestasis':3,'Drug Reaction':4,
'Peptic ulcer diseae':5,'AIDS':6,'Diabetes ':7,'Gastroenteritis':8,'Bronchial
Asthma':9,'Hypertension ':10,
'Migraine':11,'Cervical spondylosis':12,
'Paralysis (brain hemorrhage)':13,'Jaundice':14,'Malaria':15,'Chicken
pox':16,'Dengue':17,'Typhoid':18,'hepatitis A':19,
'Hepatitis B':20,'Hepatitis C':21,'Hepatitis D':22,'Hepatitis E':23,'Alcoholic
hepatitis':24,'Tuberculosis':25,
'Common Cold':26,'Pneumonia':27,'Dimorphic hemmorhoids(piles)':28,'Heart
attack':29,'Varicose veins':30,'Hypothyroidism':31,
'Hyperthyroidism':32,'Hypoglycemia':33,'Osteoarthristis':34,'Arthritis':35,
'(vertigo) Paroymsal  Positional Vertigo':36,'Acne':37,'Urinary tract
infection':38,'Psoriasis':39,
'Impetigo':40}},inplace=True)

X1= dataframe[list11]

y1 = dataframe[["prognosis"]]

nump.ravel(y1)

# TRAINING DATA
train=panda.read_csv("Testing.csv")
train.replace({'prognosis':{'Fungal infection':0,'Allergy':1,'GERD':2,'Chronic
cholestasis':3,'Drug Reaction':4,
'Peptic ulcer diseae':5,'AIDS':6,'Diabetes ':7,'Gastroenteritis':8,'Bronchial
Asthma':9,'Hypertension ':10,
'Migraine':11,'Cervical spondylosis':12,
'Paralysis (brain hemorrhage)':13,'Jaundice':14,'Malaria':15,'Chicken
pox':16,'Dengue':17,'Typhoid':18,'hepatitis A':19,
'Hepatitis B':20,'Hepatitis C':21,'Hepatitis D':22,'Hepatitis E':23,'Alcoholic
hepatitis':24,'Tuberculosis':25,
'Common Cold':26,'Pneumonia':27,'Dimorphic hemmorhoids(piles)':28,'Heart
attack':29,'Varicose veins':30,'Hypothyroidism':31,
'Hyperthyroidism':32,'Hypoglycemia':33,'Osteoarthristis':34,'Arthritis':35,
'(vertigo) Paroymsal  Positional Vertigo':36,'Acne':37,'Urinary tract
infection':38,'Psoriasis':39,
'Impetigo':40}},inplace=True)
```

```
Xtest= train[list11]


ytest = train[["prognosis"]]


nump.ravel(ytest)



# Module 1: Decision tree Classifier


def DecTree():


    #importing required modules
    from sklearn import tree
    from sklearn.metrics import accuracy_score


    clf2 = tree.DecisionTreeClassifier()
    # model of the tree
    clf2 = clf2.fit(X1,y1)


    # calculation of the accuracy of the model
    ypred=clf2.predict(Xtest)
    print(accuracy_score(ytest, ypred))
    print(accuracy_score(ytest, ypred,normalize=False))



    psymp = [Symp1.get(),Symp2.get(),Symp3.get(),Symp4.get(),Symp5.get()]


    for ko in range(0,len(list11)):
        for zen in psymp:
            if(zen==list11[ko]):
                list22[ko]=1



    it = [list22]


    prediction = clf2.predict(it)


    predictd=prediction[0]



    hh='no'
    for ah in range(0,len(diseases)):
        if(predictd == ah):
            hh='yes'
            break




    if (hh=='yes'):
        tf1.delete("1.0", END)
        tf1.insert(END, diseases[ah])
    else:
        tf1.delete("1.0", END)
        tf1.insert(END, "Nothing")



# Module 2: RandomForest Classifier
```

```
def rand():

    #Importing required modules
    from sklearn.ensemble import RandomForestClassifier
    from sklearn.metrics import accuracy_score

    clf5 = RandomForestClassifier()
    clf5 = clf5.fit(X1,nump.ravel(y1))



    # calculating accuracy

    ypred=clf5.predict(Xtest)
    print(accuracy_score(ytest, ypred))
    print(accuracy_score(ytest, ypred,normalize=False))



    psymp = [Symp1.get(),Symp2.get(),Symp3.get(),Symp4.get(),Symp5.get()]

    for ko in range(0,len(list11)):
        for zen in psymp:
            if(zen==list11[ko]):
                list22[ko]=1

    it = [list22]

    prediction = clf5.predict(it)
    predictd=prediction[0]



    hh='no'
    for ah in range(0,len(diseases)):
        if(predictd == ah):
            hh='yes'
            break


    if (hh=='yes'):
        tf2.delete("1.0", END)
        tf2.insert(END, diseases[ah])
    else:
        tf2.delete("1.0", END)
        tf2.insert(END, "Nothing")



# Module 3: NaiveBayes

def Naveb():

    # importing required modules
    from sklearn.metrics import accuracy_score
    from sklearn.naive_bayes import GaussianNB

    ganb = GaussianNB()
```

```
ganb=ganb.fit(X1,nump.ravel(y1))



# calculating accuracy

ypred=ganb.predict(Xtest)
print(accuracy_score(ytest, ypred))
print(accuracy_score(ytest, ypred,normalize=False))



psymp= [Symp1.get(),Symp2.get(),Symp3.get(),Symp4.get(),Symp5.get()]
for ko in range(0,len(list11)):
    for z in psymp:
        if(z==list11[ko]):
            list22[ko]=1

it = [list22]

prediction = ganb.predict(it)

predictd=prediction[0]

hh='no'
for ah in range(0,len(diseases)):
    if(predictd == ah):
        hh='yes'
        break

if (hh=='yes'):
        tf3.delete("1.0", END)
        tf3.insert(END, diseases[ah])
    else:
        tf3.delete("1.0", END)
        tf3.insert(END, "Nothing")

# Module 4: using TKInterface for Graphical user interface

cute = Tk()
cute.configure(background='#483D8B')

# var for entries
Symp1 = StringVar()
Symp1.set(None)
Symp2 = StringVar()
Symp2.set(None)
Symp3 = StringVar()
Symp3.set(None)
Symp4 = StringVar()
Symp4.set(None)
Symp5 = StringVar()
Symp5.set(None)
Name = StringVar()

# Heading

wall2 = Label(cute, justify=LEFT, text="Mini Project", fg="white",
bg="#483D8B")
wall2.config(font=("Forte", 30))
```

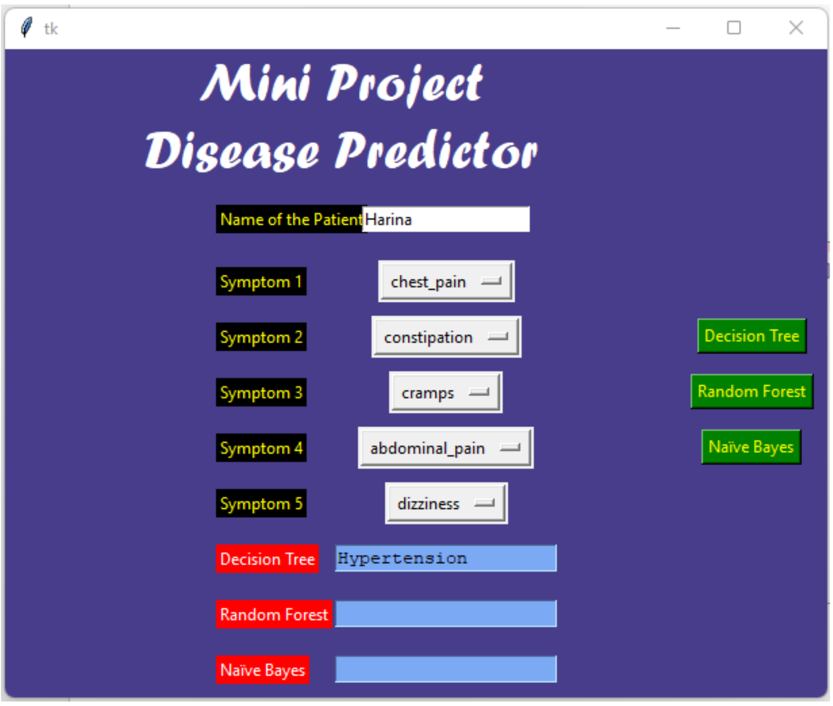
```
wall2.grid(row=1, column=0, columnspan=2, padx=100)
wall2 = Label(cute, justify=LEFT, text="Disease Predictor", fg="white",
bg="#483D8B")
wall2.config(font=("Forte", 30))
wall2.grid(row=2, column=0, columnspan=2, padx=100)

# outer
NameLable = Label(cute, text="Patient's Name", fg="yellow", bg="black")
NameLable.grid(row=6, column=1, pady=15, sticky=W)



Symp1Lb = Label(cute, text="Symptom 1", fg="#FFFF00", bg="#000000")
Symp1Lb.grid(row=8, column=1, pady=10, sticky=W)

Symp2Lb = Label(cute, text="Symptom 2", fg="#FFFF00", bg="#000000")
Symp2Lb.grid(row=9, column=1, pady=10, sticky=W)

Symp3Lb = Label(cute, text="Symptom 3", fg="#FFFF00", bg="#000000")
Symp3Lb.grid(row=10, column=1, pady=10, sticky=W)

Symp4Lb = Label(cute, text="Symptom 4", fg="#FFFF00", bg="#000000")
Symp4Lb.grid(row=11, column=1, pady=10, sticky=W)

Symp5Lb = Label(cute, text="Symptom 5", fg="#FFFF00", bg="#000000")
Symp5Lb.grid(row=12, column=1, pady=10, sticky=W)



layLb = Label(cute, text="Decision Tree", fg="white", bg="red")
layLb.grid(row=16, column=1, pady=10,sticky=W)
```

```
desLb = Label(cute, text="Random Forest", fg="white", bg="red")
desLb.grid(row=18, column=1, pady=10, sticky=W)

navfLb = Label(cute, text="Naïve Bayes", fg="white", bg="red")
navfLb.grid(row=20, column=1, pady=10, sticky=W)

# root portion for ent-ry

OPT = sorted(list11)

NameOfEntry = Entry(cute, textvariable=Name)
NameOfEntry.grid(row=6, column=1)

S1En = OptionMenu(cute, Symp1,*OPT)
S1En.grid(row=8, column=1)

S2En = OptionMenu(cute, Symp2,*OPT)
S2En.grid(row=9, column=1)

S3En = OptionMenu(cute, Symp3,*OPT)
S3En.grid(row=10, column=1)

S4En = OptionMenu(cute, Symp4,*OPT)
S4En.grid(row=11, column=1)

S5En = OptionMenu(cute, Symp5,*OPT)
S5En.grid(row=12, column=1)
```

```
dstree = Button(cute, text="DecisionTree Classifier",
command=DecTree,bg="#556b2f",fg="#FFFF00")
dstree.grid(row=9, column=2,padx=11)

rndmf = Button(cute, text="RandomForest Classifier",
command=rand,bg="#556b2f",fg="#FFFF00")
rndmf.grid(row=10, column=2,padx=11)

lrnb = Button(cute, text="Naïve Bayes",
command=Naveb,bg="#556b2f",fg="#FFFF00")
lrnb.grid(row=11, column=2,padx=11)

#project ans outlet

tf1 = Text(cute, height=1, width=20,bg="#7ca9f4",fg="#000000")
tf1.grid(row=16, column=1, padx=10)

tf2 = Text(cute, height=1, width=20,bg="#7ca9f4",fg="#000000")
tf2.grid(row=18, column=1 , padx=10)

tf3 = Text(cute, height=1, width=20,bg="#7ca9f4",fg="#000000")
tf3.grid(row=20, column=1 , padx=10)


cute.mainloop()
```

APPENDIX 3

SCREENSHOTS



Name of the Patient Harina

Symptom 1 chest_pain

Symptom 2 constipation

Symptom 3 cramps

Symptom 4 abdominal_pain

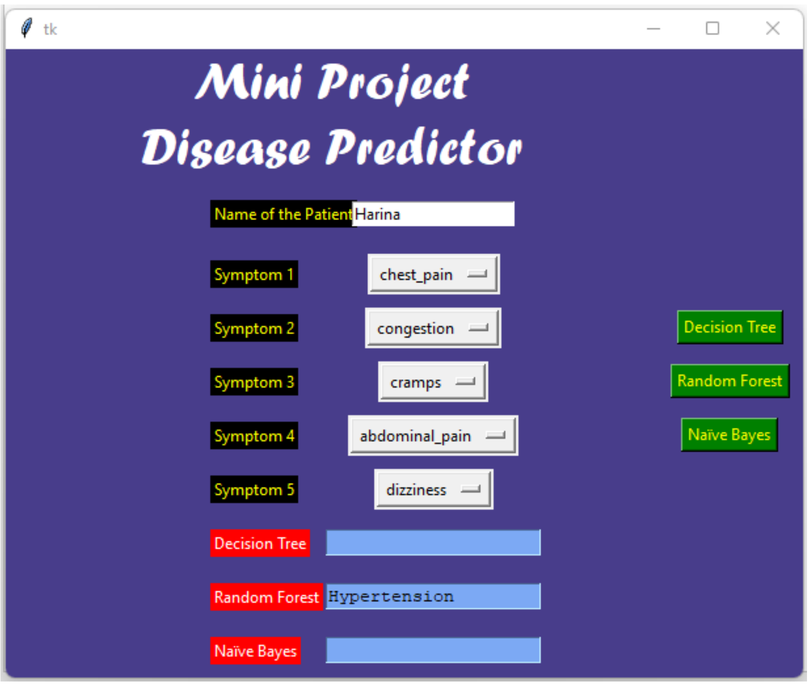
Symptom 5 dizziness

Decision Tree Hypertension

Random Forest

Naïve Bayes

FigA3.1 Prediction using Decision Tree



Name of the Patient Harina

Symptom 1 chest_pain

Symptom 2 congestion

Symptom 3 cramps

Symptom 4 abdominal_pain

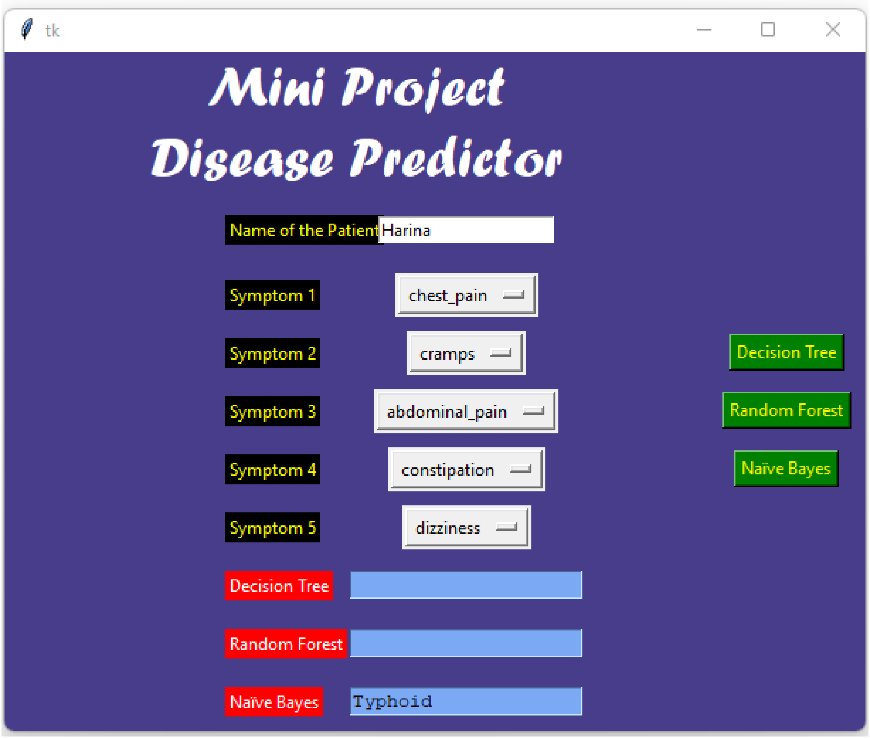
Symptom 5 dizziness

Decision Tree

Random Forest Hypertension

Naïve Bayes

FigA3.2 Prediction using RandomForest



Mini Project
Disease Predictor

Name of the Patient Harina

Symptom 1 chest_pain

Symptom 2 cramps

Symptom 3 abdominal_pain

Symptom 4 constipation

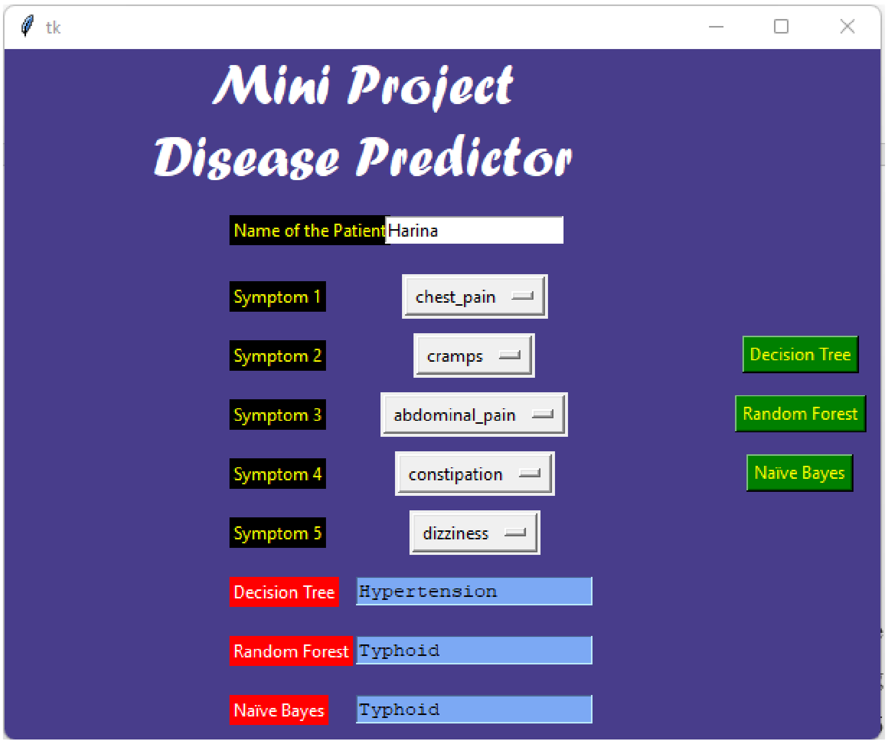
Symptom 5 dizziness

Decision Tree

Random Forest

Naïve Bayes Typhoid

FigA3.3 Prediction using Naïve Bayes



Mini Project
Disease Predictor

Name of the Patient Harina

Symptom 1 chest_pain

Symptom 2 cramps

Symptom 3 abdominal_pain

Symptom 4 constipation

Symptom 5 dizziness

Decision Tree Hypertension

Random Forest Typhoid

Naïve Bayes Typhoid

FigA3.4 Application Output

REFERENCES

[1] Dhiraj Dahiwade; Gajanan Patle; Ektaa Meshram,“ Designing Disease Prediction Model Using Machine Learning Approach”, 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC), ISBN: 978-1-5386-7809-1, 29 August 2019.

[2] G. Battineni, G. G. Sagaro, N. Chinatalapudi, and F Amenta, “Applications of machine learning predictive models in the chronic disease diagnosis”, pubmed, 2020 Mar 31.

[3] Shanjida Khan Maliha, Romana Rahman Ema, Simanta Kumar Ghosh, Helal Ahmed, Md. Rafsun Jony Mollick , “Cancer Disease Prediction Using Naive Bayes, K-Nearest Neighbor and J48 algorithm”, 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), ISBN: 978-1-5386-5907-6, 30 December 2019.

[4] K Prajwal; Tharun K; Navaneeth P; Anand Kumar M,” Cardiovascular Disease Prediction Using Machine Learning”, 2022 International Conference on Innovative Trends in Information Technology (ICITIIT),ISBN: 978-1-6654-0671-0, 01 April 2022.

[5] Muhammad Bilal Butt, Majed Alfayad, Nouh Sabri Elmitwally, “Diagnosing the Stage of Hepatitis C Using Machine Learning”, Journal of Healthcare Engineering, 2021 Dec 10.